COEN 313
Digital System Design II

Project

Section AJ-X

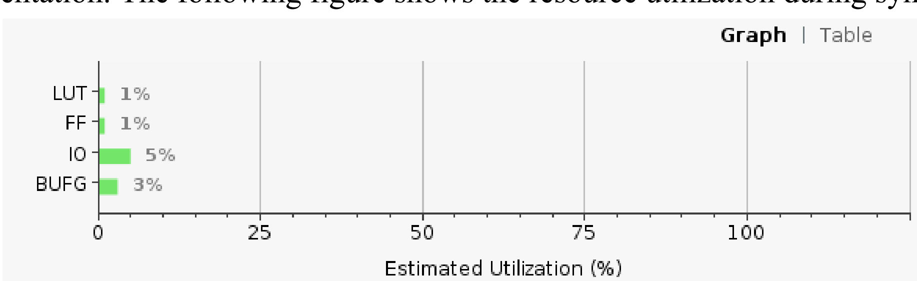
Marco Greco
40285114

Due date: June 18, 2024

The purpose of this project is to design a digital circuit that accurately tracks the occupancy of a room. When somebody enters the room, an obstruction of light would occur in a photocell, triggering a binary input signal X. When somebody exits the room, the same process occurs in a different photocell and a binary input signal Y is triggered. The maximum occupancy of the room is denoted as a 6-bit binary input signal, with a minimum of 0 and maximum of 63. The current count of people in the room is stored in a register, and the variable representing the current occupancy is updated within a clocked process. A reset signal is used to set the count of people back to 0. When the count is equal to the maximum occupancy, a digital logic output signal Z is set from 0 to 1. The circuit has been designed using a multiplexer that checks whether the current count of people is less than the maximum and if X = 1, then the count is increased. A second multiplexer checks if the current count is greater than 0 and if Y = 1, then the count is decreased. The count is denoted as a variable instead of a signal because variables are updated instantly, meaning we can account for the possibility of people entering and leaving the room at the same time. Although this event is unlikely, the measures in place to account for it will reduce the chance of bugs occurring within the system. A d flip flop is used to execute the next-output logic on every positive edge of the clock cycle. The conceptual diagram has been included in the project folder.

To simulate the code, a .do file is created which forces signals to all the inputs over several clock cycles. The .do file used for this simulation accounts for all different possible scenarios that can occur while using the occupancy tracker. First, the circuit is reset. Next, the maximum occupancy is set to 3 and, over three clock cycles, three people enter the room. The Z signal is triggered. One person then leaves the room, which sets Z back to 0, and someone reenters, setting it back to 1. Next, the count is reset again, setting Z to 0. Three people enter the room again, setting Z to 1. One person leaves, making Z equal 0, then someone enters and leaves at the same time. Z does not change. Finally, the max_occupancy is changed to 4, and the circuit is tested for the last time by having two people enter and one leave. Z is updated accordingly. Modelsim is used to perform this simulation and the waveform found in the project folder displays the results.

Finally, the code is synthesized using Xilinx Vivado. The code is successfully synthesized and implemented. The elaborated and implemented designs have been attached in the project folder, as well as the log files. No significant errors were encountered during synthesis or implementation. The following figure shows the resource utilization during synthesis.



Considering the code is quite short, the resource utilization and running time are relatively low. More concise and optimized code would lower these factors even more, however, with the small size of the project, small optimizations would be negligible. Overall, the quality of the code is high due to its concise nature and ability to function through all combinations of people entering and exiting the room without bugs. The code is easily expandable by increasing the max_occupancy, or even by adding several photocells if the room has multiple entrances or exits.

In conclusion, the occupancy tracker code was successfully written, simulated and synthesized. The low running time, low-resource code can be found in the project folder along with the conceptual diagram, the .do file, simulation results, elaborated and implemented designs, log files, originality form and README file which describes the files.